

REMARKS

Reconsideration and allowance are requested.

Claims 23-33 stand rejected under 35 U.S.C. §101. Claim 23 is amended to recite that the computer program product is embodied in a storage medium, and therefore, claim 23 recites a tangible product. Withdrawal of this rejection is requested.

The Examiner objects to the form of the claim preambles. The suggested changes have be made. Withdrawal of the objections is requested.

All claims stand rejected as being anticipated by GB Patent 2,289,354 to Jaggar—commonly assigned to ARM Ltd and submitted by Applicants. This rejection is respectfully traversed.

To establish that a claim is anticipated, the Examiner must point out where each and every limitation in the claim is found in a single prior art reference. *Scripps Clinic & Research Found. v. Genentec, Inc.*, 927 F.2d 1565 (Fed. Cir. 1991). Every limitation contained in the claims must be present in the reference, and if even one limitation is missing from the reference, then it does not anticipate the claim. *Kloster Speedsteel AB v. Crucible, Inc.*, 793 F.2d 1565 (Fed. Cir. 1986). As will be explained below, Jaggar is missing at least one claim limitation in all the claims.

The technology in this case relates to a data processing system having multiple instruction sets and the way in which such multiple instruction sets may be encoded. A data processing system supporting multiple instruction sets allows considerable flexibility in the way in which program operations may be represented. For example, that flexibility can yield improved code density. But the typical trade-off is increased hardware to support the multiple instruction sets. The inventors overcame this problem by arranging the encoding of the

instruction sets such that a common subset of instructions has the same encoding, (after variations due to storage order, e.g. endianness, have been compensated for). As a result, the data processing system implementation can be advantageously simplified. As one example, common decoding logic may be more readily utilized, thereby reducing the hardware overhead needed.

Jaggar describes multiple instruction set mapping. Program instruction words of a second instruction set are converted into or "mapped" to program instructions of a first instruction set before being decoded by a first instruction set decoder. This approach is acknowledged by the inventors as prior art and is described in conjunction with Figure 7 of the instant application. But instruction conversion is not what is claimed in this case. Instead, claim 1 recites "a subset of program instructions of said first instruction set having a common storage order compensated encoding with a subset of program instructions of said second instruction set and forming a common subset of instructions representing at least one class of instructions." The other independent claims recite an analogous feature. Jaggar does not teach this feature.

The Examiner considers this quoted feature to be anticipated by page 4, lines 24-29 of Jaggar, which reads "mapping Q bits of a Y-bit program instruction word of a second instruction set passing along said instruction pipeline to said P bits of a corresponding X-bit program instruction word for decoding by said decoding means." Here, Jaggar is describing converting 16-bit Thumb instructions to 32-bit ARM instructions before they are decoded as ARM instructions. See page 12, line 30 to page 13, line 8 and Figure 7 of this application.

In contrast to Jaggar, the two instruction sets in claim 1 share a common encoding for a common subset of instructions. As a result, the conversion taught by Jaggar is not needed. Instead, only the storage order of the program instructions need to be compensated for, since this

storage order of program instructions from the different instruction sets may vary, e.g., due to endianness differences, instruction word size differences, and the like. Consequently, the hardware which must be provided to handle instructions of the two instruction sets is significantly simplified.

It is not clear whether the Examiner is interpreting "having a common storage order compensated encoding" as being the same as converting an instruction with a first instruction encoding into a second, different instruction encoding. But such an interpretation would not be reasonable. The Examiner's attention is directed to Figure 5 in Jaggar which details a conversion from a Thumb instruction encoding into an ARM instruction encoding. The Thumb and ARM instructions shown there clearly do not "hav[e] a common storage order compensated encoding." For one thing, the Thumb instruction is half the size of the ARM instruction shown in Figure 5, so there clearly is no common encoding. Moreover, the Thumb instruction is re-encoded in order to generate the equivalent ARM instruction. For example, the value of the 4th most significant bit ("0") in the Thumb instruction at the top of Figure 5 is changed in the instruction encoding conversion to a different value ("1") in the ARM instruction (follow the "remapping" dashed line). Two instructions with different encodings require different instruction decoders. Such instruction re-encoding described and illustrated in Jaggar is not the same as compensating for storage order as claimed.

The dependent claims are distinguished from Jaggar because the independent claims distinguish from Jaggar. So no additional distinction is required. But Applicants will address an assumption the Examiner is making with regard to claim 3 and its analogous other dependent claims. Sharing registers between instruction sets in no way means or suggests that a class of instructions is shared.

SEAL, D. et al.
Appl. No. 10/781,883
August 1, 2006

The application is in condition for allowance. An early notice to that effect is respectfully requested.

Respectfully submitted,

NIXON & VANDERHYE P.C.

By: 

John R. Lastova
Reg. No. 33,149

JRL:maa
901 North Glebe Road, 11th Floor
Arlington, VA 22203-1808
Telephone: (703) 816-4000
Facsimile: (703) 816-4100